

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №1
по дисциплине «АиСД»
Тема: Рекурсия

Студент гр. 8304

Порывай П.А

Преподаватель

Фирсов М.А

Санкт-Петербург

2019

Цель работы.

Умение пользоваться рекурсивными функциями, написать программу по алгоритму описанному ниже.

Задание

Вариант 19

$\Phi(a) = a$, если $\|a\| \leq 2$, $\Phi(a) = \Phi(b)\Phi(y)$ если $a = by$ $\|b\| = \|y\|$ $\|a\| > 2$,

$\Phi(a) = \Phi(bh)\Phi(hy)$ если $a = bhy$ $\|b\| = \|y\|$, $\|a\| > 2$, $\|h\| = 1$

Выполнение работы

Функция RecF принимает на вход вектор целых чисел и применяет к нему алгоритм описанный в задании. При каждом вызове выводится вектор, который был в аргументе при этом выполняется отступ по глубине. Если длина вектора делится на 2 и не равна 2 нужно вызвать 2 функции RecF. Как первая так и вторая функции в конструкции вида $(length \% 2 == 0 \ \&\& \ length \neq 2) \{ \dots \}$ дают определенную глубину(в них еще могут быть вызваны функции).

Аналогично для конструкции где длина вектора делится на 2. При длине 1 или 2 часть возвращающегося вектора записывается в строку.

Функция is_alldigits проверяет вектор из чисел записанных как строка, при этом используются итераторы it1, it2 для прохода по вектору из чисел и по самим цифрам числа соответственно, если в начале числа «-», сдвигаем итератор строки вправо

В main выводится диалоговое окно предлагающее считать данные из файла или из консоли. При выборе счета из файла создается объект класса ifstream, ofstream. В while считывается строка из объекта входного потока с помощью функции getline. Далее в теле цикла если была записана пустая строка то на экран и выходной файл помещается соответствующее сообщение. Вызывается функция is_alldigits. В итоге если в исходной строке последовательность целых чисел то вызывается функция RecF. Операции при выборе счета с консоли осуществляются почти аналогично.

Выводы.

Получены навыки работы с рекурсивными функциями, ввода последовательности неизвестной длины с помощью средств C++. Также приобретены навыки работы по защите программы от неправильных входных данных.

ПРИЛОЖЕНИЕ А.ИСХОДНЫЙ КОД

```
#include<iostream>
#include<vector>
#include<iterator>//ostream_iterator..
#include<string>//getline()
#include <sstream>//istringstream()
#include<fstream>// ifstream()
#include<cctype>

using namespace std;

void RecF(vector<int>& vectorin, string* vectorout, int ident, int length)
{

    int i, j;

    for (i = 0; i < ident; i++)
        cout << "\t";

    cout << "RecF(";

    for (i = 0; i < length; i++)
        cout << vectorin[i] << " ";

    cout << ")" << endl;

    if (length % 2 == 0 && length != 2) {

        vector<int> vectorcopy1(length / 2);
        vector<int> vectorcopy2(length / 2);

        for (i = 0; i < length / 2; i++)
            vectorcopy1[i] = vectorin[i];

        RecF(vectorcopy1, vectorout, ident + 1, length / 2);
        j = 0;

        for (i = length / 2; i < length; i++) {

            vectorcopy2[j] = vectorin[i];
```

```

        j++;
    }

    RecF(vectorcopy2, vectorout, ident + 1, length / 2);

}

else if (length % 2 != 0 && length != 1) {

    vector<int> vectorcopy1((length / 2) + 1);
    vector<int> vectorcopy2((length / 2) + 1);

    for (i = 0; i < (length / 2) + 1; i++)
        vectorcopy1[i] = vectorin[i];

    RecF(vectorcopy1, vectorout, ident + 1, (length / 2) + 1);

    j = 0;

    for (i = (length / 2); i < length; i++) {

        vectorcopy2[j] = vectorin[i];
        j++;
    }

    RecF(vectorcopy2, vectorout, ident + 1, (length / 2) + 1);

}

else if (length == 2) {

    stringstream ss1;
    ss1<<vectorin[0];
    *vectorout += ss1.str();

    *vectorout += " ";

    stringstream ss2;
    ss2<<vectorin[1];
    *vectorout += ss2.str();
}

```

```

        *vectorout += " ";

    }

    else if (length == 1) {
        stringstream ss1;
        ss1<<vectorin[0];
        *vectorout += ss1.str();
    }

}

int is_alldigits(vector<string> is_digits){//передае вектор из чисел

    vector <string> :: iterator it1= is_digits.begin();
    int flag = 1;

    while(it1 != is_digits.end() ){

        string::iterator it2 = (*it1).begin();

        if(*it2 == '-')
            it2++;

        while(it2 != (*it1).end() && isdigit(*it2) ) ++it2;

        if(it2 != (*it1).end()){
            flag = 0;
            break;
        }

        it1++;
    }
    return flag;
}

int main(int argc, char* argv[]) {

```

```

setlocale(LC_ALL, "Russian");
string vector_out;
vector_out = "";
int ident = 0;
int flag;

cout << "Ввод из файла или из консоли? (f , c)\n";

char arg;
cin >> arg;

if (arg == 'f') {

    ifstream data(argv[1]);
    ofstream fout("out.txt");

    if(data){

        string str;

        while (getline(data, str)) {

            if(str == ""){

                cout<<"Пустая строка"<<endl;
                fout<<"Пустая строка"<<endl;
                continue;

            }

            istringstream s1{ str };
            vector<string> is_digits;
            is_digits.assign(istream_iterator<string>{s1}, {});

            flag = is_alldigits(is_digits);

            if(flag == 0){

                cout<<"Данные во вновь обрабатываемой строке
введены некорректно"<<endl;

                fout<<"Данные во вновь обрабатываемой строке
введены некорректно";

```

```

        fout<<"\n";
    //    break;
    }
    else{
        istreamstringstream s2{ str };
        vector<int> vector_inp;
        vector_inp.assign(istream_iterator<int>{s2},
{ }); //istream_iterator описывает объект итератора ввода, assign в классе vector
обобщенная функция
        RecF(vector_inp,      &vector_out,      ident,
vector_inp.size());

        cout << vector_out << endl;
        fout<<vector_out;
        fout<<"\n";

        vector_out = "";
    }
}

fout.close();
data.close();
}
else
    cout<<"Файл не открыт"<<endl;

}

else if (arg == 'c') {

    string str;
    getline(cin, str);

    ofstream fout("out.txt");

    while (getline(cin, str)) {

        if(str == ""){

            cout<<"Пустая строка"<<endl;
            fout<<"Пустая строка"<<endl;
            continue;
        }
    }
}

```



```

        istream s1{ str };
        vector<string> is_digits;
        is_digits.assign(istream_iterator<string>{
s1}, {}));

        flag = is_alldigits(is_digits);

        if(flag == 0){

            cout<<"Данные во вновь
обрабатываемой строке введены некорректно"<<endl;
            fout<<"Данные во вновь обрабатываемой строке
введены некорректно"<<endl;

            // break;
        }
        else{

            istream s2{ str };
            vector<int> vector_inp;
            vector_inp.assign(istream_iterator<int>{s2},
{}));

            RecF(vector_inp,      &vector_out,      ident,
vector_inp.size());

            cout << vector_out << endl;
            fout<<vector_out;
                fout<<"\\n";

            vector_out = "";
        }

    }

    fout.close();
    // data.close();

}

else
    cout << "Нет такой команды";

```

```
        return 0;  
  
    }
```

Тесты

Ввод

```
1 2 3  
12 32 43 21  
2123 21 31 1  
1 1 2 3 4  
1 -2 3 4 5  
fdsdf  
123.123 123 12wq  
1 2 3
```

Вывод

```
1 2 2 3  
12 32 43 21  
2123 21 31 1  
1 1 1 2 2 3 3 4  
1 -2 -2 3 3 4 4 5  
Данные во вновь обрабатываемой строке введены некорректно  
Данные во вновь обрабатываемой строке введены некорректно  
1 2 2 3
```